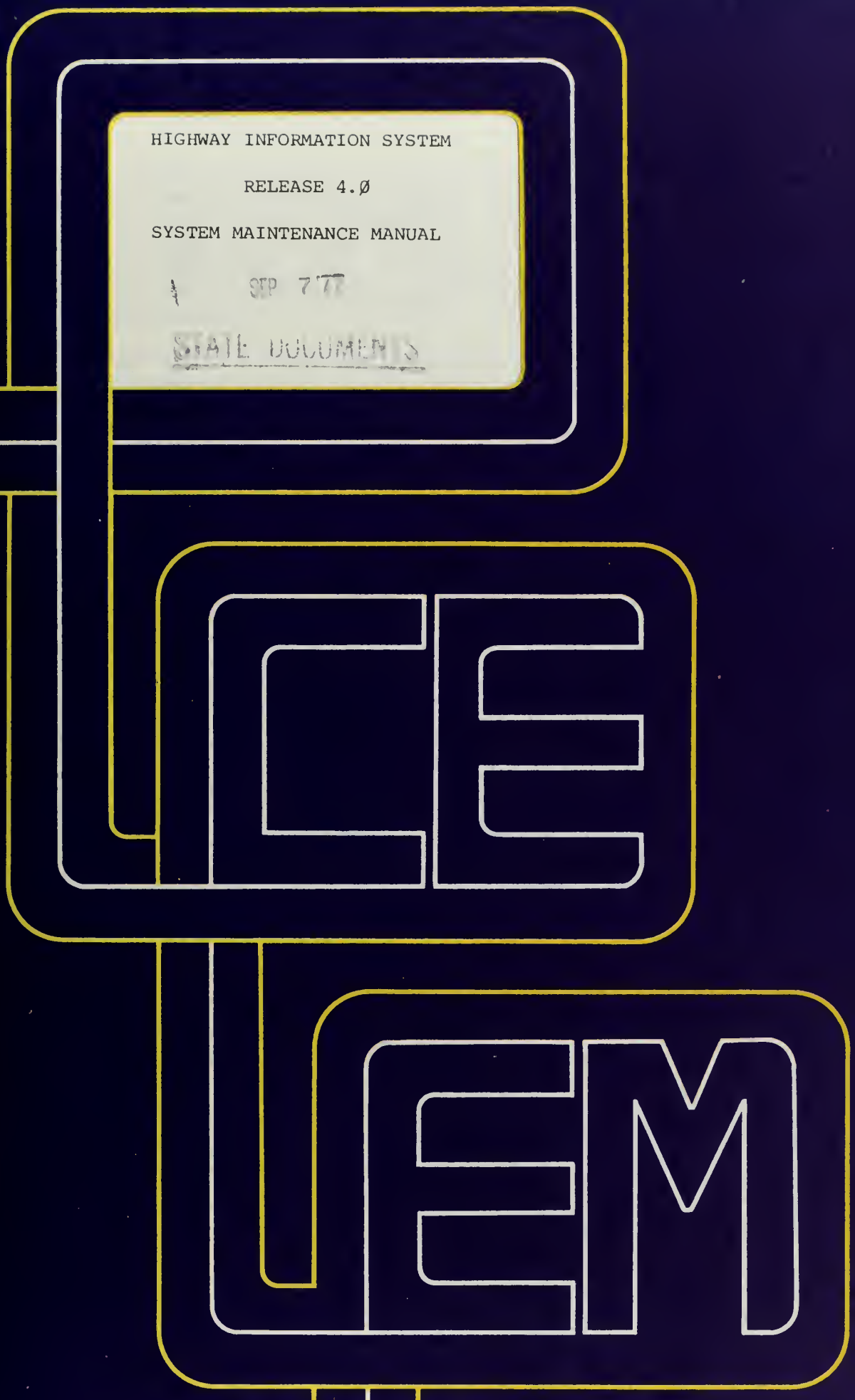
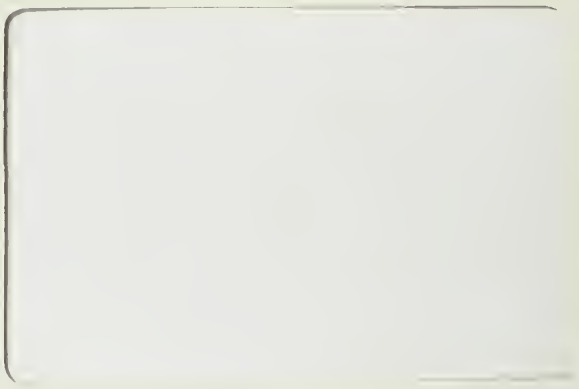
HIGHWAY INFORMATION SYSTEM

RELEASE 4.Ø

SYSTEM MAINTENANCE MANUAL



Department of Civil Engineering & Engineering Mechanics Montana State University Bozeman



HIGHWAY INFORMATION SYSTEM

RELEASE 4.Ø

SYSTEM MAINTENANCE MANUAL

Prepared for the:

STATE OF MONTANA
DEPARTMENT OF HIGHWAYS
PLANNING AND RESEARCH BUREAU

In cooperation with the:

U.S. DEPARTMENT OF TRANSPORTATION
FEDERAL HIGHWAY ADMINISTRATION

Project Director: Ralph W. Zimmer, P.E.

Prepared by:

Larry J. Coats, Edward G. Knoyle, and Ralph W. Zimmer

DEPARTMENT OF CIVIL ENGINEERING AND ENGINEERING MECHANICS
MONTANA STATE UNIVERSITY
Bozeman, Montana 59715

April, 1976

FOREWORD

This report is a portion of the documentation of Release 4.Ø of the Highway Information System undertaken by the Department of Civil Engineering and Engineering Mechanics, Montana State University. The retrieval system has been evolving over the last several years under the sponsorship of the Planning and Research Bureau of the Montana Department of Highways with some assistance from the Highway Traffic Safety Division, Montana Department of Community Affairs.

Release 4.Ø of the Highway Information System is documented in the following volumes:

Highway Information System Release 4.Ø: System Overview

Provides an introduction to the Highway Information System.

Highway Information System Release 4.Ø: Index

Provides an index to all manuals except the System Overview and Program Listings.

Highway Information System Release 4.Ø: User's Manual

Describes how to use the Highway Information System for retrieving information and for printing reports and summaries.

Highway Information System Release 4.Ø: Data Coding Manual

Describes the data card formats for entering data into the Highway Information System files.

Highway Information System Release 4.Ø: System Maintenance Manual

Provides information for performing scheduled system backups and file reorganizations and for allocating system files.

Highway Information System Release 4.Ø: Record Formats & Subroutines

Describes the internal record formats of the various files and provides calling sequences to subroutines. This manual is intended for persons writing new programs to add to the Highway Information System.

Highway Information System Release 4.Ø: Programming Details

Describes the existing programs and provides a guide to the program listings. This manual is intended for persons maintaining existing software in the Highway Information System.

Highway Information System Release 4.Ø: Program Listings

Contains computer-generated listings of all source programs of the Highway Information System.

Although the project was conceived, initiated, and primarily funded through the Planning and Research Bureau of the Montana Department of Highways, the development cost of selected portions of the system was borne by the Highway Traffic Safety Division of the Montana Department of Community Affairs.

In developing the system, the CE & EM Department has had the privilege of using an IBM OS/VS1 37Ø/145 computer located at the Data Processing Bureau of the Montana Department of Highways in Helena. PL/I has been used for most of the programs because of its versatility and ease of use. BAL (assembler) has been used for most input-output modules and for other modules that require its increased capabilities and efficiency over PL/I.

The project could never have progressed to its current state without the continued and patient encouragement and assistance from the Planning and Research Bureau and the Data Processing Bureau of the Montana Department of Highways, and from the Highway Traffic Safety Division of the Department of Community Affairs.

The project conclusion was also hastened by the significant effort of other project personnel: Scott H. Danforth, R. Helene Knowlton, and Doug M. Geiger.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

This manual is intended for use by those persons involved in the maintenance of the Highway Information System's files. It is also intended for use by persons involved in the allocation of Highway Information System files.

The manual has the following structure:

Chapter 1 - Introduction.

Chapter 2 - Describes the file organizations used in the Highway Information system.

Chapter 3 - Describes the use of several utility programs provided for use by maintenance personnel.

Chapter 4 - Describes the on-going system maintenance that is required. Included in this chapter are instructions for system backups, file reorganization, and the creation of directory and report files.

Chapter 5 - Provides information pertaining to the allocation of Highway Information System files.

CHAPTER 2

FILE ORGANIZATIONS

The following file organizations are used in the Highway Information System:

1. Sequential organization - Used for small files in which access via a "key" is not needed.
2. Partitioned organization - Used for object modules, load modules, and tables.
3. Indexed-sequential organization - Used for data files in which access via a "key" is needed.
4. Virtual Storage Access Method - VSAM files provide an alternative to indexed-sequential organization.

This chapter describes the features of these file organizations that are used in HIS.

Sequential Organization

Sequential files are the simplest files. A sequential file consists of a set of records that can be read in order from the first record in the file to the last. Direct access into a sequential file is generally not feasible. Sequential files can reside on all storage units, including disks and tapes. Card files and printer files are sequential files. The other file organizations discussed in this chapter can reside only on disk storage units.

Sequential files are used for system backups and for several small files in which direct access is not needed.

When allocating a new file on tape that is to be kept, a DD statement of the following format must be included with the run:

```
//ddname DD UNIT=TAPE,VOL=SER=serial,DISP=(NEW,KEEP),DSNAME=name,
//          LABEL=(sequence,RETPD=retpd),
//          DCB=(BLKSIZE=blksize,LRECL=lrecl,RECFM=FB)
```

UNIT=TAPE specifies that the file is a tape file. VOL=SER=serial specifies the serial number of the tape. DISP=(NEW,KEEP) specifies that the file is a new file that is to be kept after the job ends. DSNAME=name specifies a name by which the file will be known in the future. LABEL=(sequence,RETPD=retpd) specifies both

the sequence number of the file (e.g., a sequence value of 3 specifies that the file is the third file on the tape) and the retention period (e.g., a retpd value of 365 specifies that the file will be retained 365 days). The DCB parameter specifies information about how the information is stored in the file. RECFM=FB specifies fixed-length blocked records. LRECL=lrecl specifies the length of the records (logical records). BLKSIZE=blksize specifies the length of the blocks (physical records), and must be a multiple of lrecl. In general, a large block size should be chosen for tape files (the maximum value is 32ØØØ).

The following DD statement provides access to an existing tape file:

```
//ddname DD UNIT=TAPE,VOL=SER=serial,DISP=OLD,DSNAME=name,
//          LABEL=sequence
```

For files on disk packs, it is necessary to estimate the number of records that will be stored in the file in order to determine the amount of space that must be allocated. It is also recommended that a block size be calculated to provide good use of disk space. The CALC-BLOCKSIZE program documented in chapter 3 can aid in calculating block sizes. The calculations used for calculating block sizes are:

1. Decide how many physical records (blocks) should be stored on each track of the disk pack. The best use of disk storage is with one block per track (full track blocking), but this requires more storage in core for buffers when running a program that uses the file.
2. IBM provides a table that gives the maximum block size for a given number of blocks per track. A portion of this table is reproduced here:

Blocks/Track	333Ø	2314
1	13Ø3Ø	7294
2	6447	3521
3	4253	2298
4	3156	1693
5	2498	1332
6	2Ø59	1Ø92
7	1745	921
8	151Ø	793
9	1327	694
1Ø	1181	615

Choose the appropriate value from this table.

3. Divide the value found in step 2 by the logical record length and drop the remainder. This value is the number of logical records per block.

4. Multiply the value calculated in step 3 by the logical record length. This value is the block size.

For example, suppose you are allocating a file on 333Ø disk that has a logical record length of 1ØØ. You wish to have four blocks on each track. From the table lookup, you find that your maximum block size is 3156. Dividing by 1ØØ gives you a value of 31 records per block. Your block size is 31ØØ. Since four blocks will be stored on each track, 124 records can be stored on each track. Use of storage could be maximized by using full track blocking (this would result in 13Ø records per track). A great deal of storage can be wasted by choosing an inappropriate block size. For example, suppose a block size of 7ØØØ was chosen. With this block size, only one block can be written on each track, and only 7Ø records will be stored on each track.

To calculate the number of records that will be stored on a cylinder, multiply the number of records stored per track by 19 (333Ø disks) or by 2Ø (2314 disks). Estimate the number of records that must be stored in the file to determine how many cylinders (or tracks) must be allocated.

When allocating a file on disk that will be retained, use the following DD statement:

```
//ddname DD UNIT=unit,VOL=SER=serial,DISP=(NEW,disp),SPACE=(CYL,n),
//          DSNAME=name,DCB=(RECFM=FB,LRECL=lrecl,BLKSIZE=blksize)
```

UNIT=unit is an installation-dependant parameter. On the Montana Department of Highways' computer, specify DISK for 2314, 333Ø for single-density 333Ø, or 333ØP for dual-density 333Ø. Code the serial number of the disk pack in the VOL=SER=serial parameter. Code KEEP or CATLG in the DISP parameter. Code the number of cylinders in the SPACE parameter (or code number of tracks as SPACE=(TRK,n)). The DSNAME parameter specifies the name by which the file will be known. The DCB parameter specifies record length and block size information. To process a previously-cataloged disk file, use the following DD statement:

```
//ddname DD DISP=OLD/SHR,DSNAME=name
```

If the file is not cataloged, UNIT and VOL=SER must also be included in the DD statement.

Partitioned Organization

A partitioned file is essentially a set of several sequential files. Each of these sequential files has a member name by which it can be referenced. A partitioned file is easier to work with than is a number of separate sequential files because substantially less Job Control Language is needed - a single DD statement provides access to all of the members in the file. All members of a partitioned data set must have the same record format, record length, and block size.

When a member of a partitioned data set (or library) is updated, a fresh version is written to the file and the old version is deleted. The space used by the old version is not made available for reuse. For this reason, the library becomes full sooner or later and must be reorganized to free the deleted space. A library that is updated frequently must be allocated with enough space so that the library does not become full too quickly. A cataloged library can be reorganized by using the following job setup:

```
// JOB
// EXEC COMPRESS,LIBRARY='dsname'
```

A portion of the library is reserved for a "directory" that provides member names and locations. When a library is allocated, the DD statement must specify the number of directory blocks that are to be allocated. The more blocks specified, the more members can be stored in the library. The number of members that can be described in a directory block depends upon the information that is stored in the block. For a load module library, one directory block can describe about 5 or 1Ø members. For other libraries in which no user fields are added, one directory block can describe about 2Ø to 3Ø members.

When a library is allocated, the DD statement appears just like the one above for allocating a sequential disk file except that the space parameter is written in the format SPACE=(CYL,(n,,b)), where n is the total number of cylinders and b is the number of directory blocks. For load modules, the DCB parameter can be omitted. The DD statement for accessing an existing library is the same as that for accessing an existing sequential file on disk.

Optimum block sizes for libraries are calculated in the same manner as for sequential files.

Indexed-Sequential Organization

An indexed-sequential file is a "souped-up" sequential file. It provides an indexing scheme that allows records to be accessed on the basis of the contents of a "key" field, as well as to be accessed sequentially. It also provides features that allow records to be added to the file and to be deleted from the file.

The Job Control Language for allocating indexed-sequential files is considerably more complicated than that for allocating sequential files. Before a method of allocating the file is chosen, you must consider the use of the file:

1. Will records be added to the file at any time, and if so will the update rate be low, moderate, or high?
2. Will records be deleted from the file?
3. Will most access to the file be direct or sequential?

Overflow areas for inserting records are of two types: "cylinder" overflow areas and "independant" overflow areas. Cylinder overflow areas are the most efficient in terms of processing speed because disk arm movement is minimized. When using cylinder overflow, a portion of each cylinder is reserved for overflow records. When using independant overflow, a separate area is allocated solely for overflow records.

If a file will have a low or moderate update rate (or will not be updated at all), it may be best to use simply cylinder overflow. The DD statement for this type of file is reasonably simple:

```
//ddname DD UNIT=unit,VOL=SER=serial,DISP=(NEW,disp),DSNAME=name,
//          SPACE=(CYL,(n,,1)),DCB=(DSORG=IS,KEYLEN=keylen,RKP=rkp,
//          RECFM=FB,LRECL=lrecl,BLKSIZE=blksize,OPTCD=LYR,CYLOFL=n)
```

UNIT, VOL=SER, DISP, and DSNAME are the same as for sequential files. The SPACE parameter specifies the number of cylinders needed for storing the data and also specifies that one cylinder is needed for an index. DSORG=IS specifies indexed-sequential organization. KEYLEN specifies the length of the key field, and RKP specifies the relative key position (RKP=1 specifies that the key field starts in the second byte of the record, since the first byte is reserved to flag deleted records). OPTCD=LYR specifies that cylinder overflow is used and that records can be deleted. CYLOFL=n specifies the number of tracks on each cylinder that are reserved for overflow records.

If a file has a high update rate, both cylinder overflow and independant overflow should be used. The cylinder overflow is chosen by the operating system, but the independant overflow provides additional storage areas for use when a cylinder overflow area becomes full. Independant overflow is allocated separately from the prime data area. The index area can also be allocated separately. The following example shows separate allocation for each of the areas:

```
//ddname DD UNIT=unit,VOL=SER=serial,DISP=(NEW,KEEP),DSNAME=name(INDEX),
//           SPACE=(CYL,1),DCB=(DSORG=IS,KEYLEN=keylen,RKP=rkp,
//           RECFM=FB,LRECL=lrecl,BLKSIZE=blksize,OPTCD=LIYR,CYLOFL=n)
//       DD UNIT=unit,VOL=SER=serial,DISP=(NEW,KEEP),SPACE=(CYL,n),
//           DSNAME=name(PRIME),DCB=DSORG=IS
//       DD UNIT=unit,VOL=SER=serial,DISP=(NEW,KEEP),SPACE=(CYL,n),
//           DSNAME=name(OVFLOW),DCB=DSORG=IS
```

OPTCD=LIYR specifies that records can be deleted from the file and that both types of overflow areas are used.

Calculation of optimum block sizes is similar to that for sequential files. The maximum block size table is slightly different:

Blocks/Track	333Ø	2314
1	12974	7249
2	6391	3476
3	4197	2254
4	31ØØ	1649
5	2442	1288
6	2ØØ3	1Ø49
7	1689	878
8	1454	75Ø
9	1271	65Ø
1Ø	1125	571

After looking up the maximum block size from this table, the key length must be subtracted. As an example, assume you are allocating a file that has a record length of 1ØØ and a key length of 15. You wish to store 4 blocks per track. The maximum block size (for 333Ø disks) is 31ØØ - 15, or 3Ø85. The number of records per block is 3Ø85 / 1ØØ, or 3Ø. The resulting blocksize is 3ØØØ, which allows 12Ø records to be stored on each track. The number of prime tracks per cylinder is 18-n (333Ø) or 19-n (2314), where "n" is the number of tracks reserved for

cylinder overflow (note that one track per cylinder is reserved for indexes). For this file, full track blocking would result in (12974-15)/1ØØ or 129 records per track.

If a file is to be used primarily for sequential access, the choice of a large block size will significantly improve performance. Note that additional core will be needed for buffers when a larger block size is used. If a file is to be used primarily for direct access, best performance will be obtained with unblocked records or with a small block size. If the file is to be used for both sequential and direct access frequently, choose a moderate block size such that about four blocks will be stored on each track.

Indexed-sequential files need periodic reorganization. This is for several reasons:

1. Overflow areas can eventually become full.
2. Space used by deleted records is often not made available for reuse.
3. A file that has a number of records in overflow areas causes loss of efficiency in programs that use the file.

Reorganization is accomplished by either re-loading the file from a backup copy or by using the IEBISAM utility program to copy the file to a new area on disk and re-cataloging. The first method is used in the Highway Information System so that the files do not have to be re-allocated each time reorganization is performed.

VSAM Organization

VSAM is not yet being used in the Highway Information System, but some thought has been given to utilizing VSAM for some future files. VSAM provides all of the facilities of indexed-sequential organization plus several additional features. However, VSAM also increases overhead costs by drastically increasing core storage requirements for programs.

CHAPTER 3

UTILITY PROGRAMS

This chapter describes the use of several utility programs provided with the Highway Information System. These programs are useful to persons maintaining the HIS files and to persons allocating new files.

The CALC-BLOCKSIZE Program

CALC-BLOCKSIZE calculates optimum block sizes for indexed-sequential and physical-sequential files. It computes the optimum block sizes for blocking factors 1 through 1Ø on both 333Ø and 2314 disk packs. For each blocking factor printed, the following information is shown:

Block size
Number of records in each block
Number of records on each track
Number of records on each cylinder

If the program is provided with an estimated number of records, the program also prints the minimum number of cylinders that must be allocated (for indexed-sequential files, the number printed is the number of prime cylinders that are needed).

To use CALC-BLOCKSIZE, submit the following run:

```
// JOB
// EXEC HIS
//SYSIN DD *
:CALC-BLOCKSIZE
/*
//BLKSIZE DD *
  --- one or more data cards ---
/*
```

Data cards are prepared in the following format:

Columns	Length	Contents
1-2	2	File type - IS (indexed-seql) or PS (physical-seql)
3	1	Blank
4-7	4	Record length - code leading zeroes
8	1	Blank
9-1Ø	2	PS - blank IS - Key length (code leading zeroes)
11	1	Blank

Columns	Length	Contents
12-13	2	PS - Blank IS - CYLOFL (optional)
14	1	Blank
15-22	8	Number of records (optional) - code leading zeroes
23-8Ø	58	Blank or comments

The CYLOFL (cylinder overflow) value in columns 12-13 affects the number of records stored on each cylinder for indexed-sequential file. The value is the number of tracks of each cylinder reserved for overflow records.

The PRINT-DSCB Program

PRINT-DSCB prints selected items from a Data Set Control Block (DSCB). A DSCB is the control block stored on a disk pack that describes a file.

The output from PRINT-DSCB is most useful for indexed-sequential files. The output can help decide when a file needs to be reorganized and can warn when a file is nearly full.

To use PRINT-DSCB, submit the following run:

```
// JOB
// EXEC HIS
//ddname DD DISP=SHR,DSNAME=dsname
//SYSIN DD *
:PRINT-DSCB,DDNAME=ddname
/*
```

The DD statement "ddname" identifies the file for which the DSCB print-out is desired.

The output from PRINT-DSCB is in two parts. The first part shows general information about the file and has the following format:

```
PRINT-DSCB       mmm dd,yyyy

DDNAME=ddname  ---------------------------- ddname as coded in command
DSNAME=dsname  ---------------------------- dsname as coded in DD stmt
VOL=SER=serial ---------------------- serial number of disk pack

----- DCB FIELDS -----

  DSORG=IS - INDEXED-SEQUENTIAL
  BLKSIZE=n  ---------------------------------------- block size
  LRECL=n    ---------------------------------------- record length
  RECFM=x    ---------------------------------------- record format
  OPTCD=x    ---------------------------------------- option code(s)
  KEYLEN=n   ---------------------------------------- key length
  RKP=n      ---------------------------------- relative key position
  CYLOFL=n   ---------------------------------- cylinder overflow
  NTM=n      ---------------------------------- tracks for master index
```

The second part provides detailed information about the file's status. This part is broken down into seven sub-divisions. A description of each of these follows:

```
----- FILE STATUS -----

a    NUMBER OF RECORDS IN PRIME AREA       n    x % OF TOTAL
b    NUMBER OF RECORDS IN OVERFLOW AREA    n    x % OF TOTAL
c    TOTAL                                 n
d    NUMBER OF DELETED RECORDS             n
e    TOTAL NUMBER OF RECORDS IN FILE       n
```

If a file is fully reorganized, lines a, c, and e will show the same number of records and lines b and d will show zero. The percentages in lines a and b are based on the total figure in line c.

```
INDEX SPACE ALLOCATED            x TRACKS    y CYLINDERS
PRIME SPACE ALLOCATED            x TRACKS    y CYLINDERS
INDP OVERFLOW SPACE ALLOCATED    x TRACKS    y CYLINDERS
```

This portion shows the amount of space allocated for the index, the prime data, and the independant overflow.

```
a    CYLINDER OVERFLOW TRACKS PER PRIME CYLINDER    x TRACKS
b    INDEX TRACKS PER PRIME CYLINDER                1 TRACK
c    PRIME TRACKS PER PRIME CYLINDER                x TRACKS
```

Line a is the number of tracks reserved for cylinder overflow in each prime cylinder. Line b is the number of tracks reserved for index areas in each prime cylinder, and is always 1. Line c is the number of tracks of each cylinder that remain for prime data and is calculated as 19-a-b (333Ø) or as 2Ø-a-b (2314).

```
(PRIME) RECORDS/BLOCK      n
(PRIME) RECORDS/TRACK      n    x % UTILIZATION
(PRIME) RECORDS/CYLINDER   n
```

The percent utilization shown on the second line is a comparison of track usage compared with usage when using full track blocking. If full track blocking is used, the value 1ØØ% is always printed.

```
CURRENT FILE CAPACITY        x RECORDS
SPACE LEFT FOR RECORDS       x RECORDS
PERCENT OF SPACE UTILIZED    y %
```

This portion is self explanatory.

```
NUMBER OF INDEX LEVELS                x
TRACKS TO CREATE MASTER INDEX         x
HIGH-LEVEL INDEX - BYTES              x
HIGH-LEVEL INDEX - TRACKS             x
```

This portion is provided to aid in determining if and when a master index should be created for the file.

```
RECORDS/CYLINDER OVERFLOW TRACK                    x
RECORDS/INDEPENDENT OVERFLOW TRACK                 x
EMPTY TRACKS IN INDEPENDENT OVERFLOW AREA          x
FULL CYLINDER OVERFLOW AREAS                       x
```

When several cylinder overflow areas become full or when few tracks are left in the independent overflow area, the file needs to be reorganized.

The PRINT-JFCB Program

PRINT-JFCB prints the contents of a Job File Control Block (JFCB). A JFCB is the control block that contains the information specified in a DD statement. To use PRINT-JFCB, submit the following job setup:

```
// JOB
// EXEC HIS
//ddname DD ...
//SYSIN DD *
:PRINT-JFCB,DDNAME=ddname
/*
```

The COPY-SEQL-FILE Program

COPY-SEQL-FILE copies a sequential file. It can change the record length by truncating on the right or by padding with blanks on the right. Use the program with the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD ... (input file)
//OUTDD DD ... (output file)
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

If the record lengths of the two files do not match, specify TYPE-RUN=NOMATCH on the COPY-SEQL-FILE command.

The PANVALET Backup and Restore Programs

All of the Highway Information System source modules are kept within the Department of Highways' PANVALET system. In addition, they have been kept in a separate backup system for the following reasons:

1. Transportability - So that the modules can be taken to an installation that does not have the PANVALET system.
2. Documentation - Listings of modules in the backup system can be obtained easily for inclusion in the system documentation.

The PANBACK Program - Use PANBACK to perform a backup. PANBACK requires the following files:

1. Input tape from previous backup. This tape is merged with the new backup modules.
2. Input data cards. This data describes the backup functions to be performed.
3. Output tape. This tape will be the new backup.

Use the following job setup with the PANBACK program:

```
// JOB
// EXEC HIS,COPIES=n,OUTRET=365,
//      INSER=serial,INSEQ=sequence,INNAME=name,
//      OUTSER=serial,OUTSEQ=sequence,OUTNAME=name
//SYSIN DD *
:PANBACK
/*
//INPUT DD *
  --- data cards ---
/*
```

The program operates in three steps:

1. Read data cards and prepare merge file.
2. Merge the merge file with the input tape.
3. Execute PANVALET program.

If any of these steps fail, the remaining steps are not executed.

Prepare data cards with a module name in columns 1-1Ø and an option code in column 11. The option code can be any of these codes:

blank	Backup only
I	Backup and mark INACTIVE and PROD
X	Delete from input tape

These cards must be entered in order by module name. Any PANVALET commands may be interspersed with the data cards, and will be executed during the third step.

The PANRSTR Program

The PANRSTR program can be used to restore/replace modules to the active PANVALET library from the backup system. Use the following job setup:

```
// JOB
// EXEC HISPANRS
//SYSIN DD *
  ---data cards---
/*
```

Data cards can have any of these formats:

++RESTORE name

Use to restore a module to the active library when no member is already stored with the specified name.

++REPLACE name,level

Use to restore a module when a member is already stored with the specified name. "level" is the level of the member that is already stored in the active library.

$$ADD name,language,user

(alternate) Can be used to restore a module to the active library.

$$UPDATE name,level,ALL

(alternate) Can be used to replace a module in the active library.

Any PANVALET commands may be interspersed with these data cards.

The PANBACK-TO-PDS Program - This program can be used to copy modules from the backup system to a partitioned data set. Use the following job setup:

```
// JOB
// EXEC HIS
//LIBRARY DD ....  (partitioned data set)
//BACKUP  DD ...   (backup tape)
//SYSIN DD *
:PANBACK-TO-PDS
/*
//SOURCE DD *
  --- data cards ---
/*
```

Each data card contains the name of one source module in columns 1-1Ø. The cards must be entered in order by name.

Source Module Listings - The PRINT-SOURCE-LISTING program can print source modules from the backup system. Use the following job setup:

```
// JOB
// EXEC HISPANLS
//SYSIN DD *
:PRINT-SOURCE-LISTING,LIST=SEQ/NOSEQ,LOCATION=location
/*
```

LIST=SEQ requests that sequence numbers be printed. LIST=NOSEQ suppresses the sequence number.
The LOCATION parameter has two formats. LOCATION=name specifies that all modules whose name begins with "name" are printed. LOCATION=name1-name2 specifies that all modules whose name falls between name1 and name2 (inclusive) are printed.

CHAPTER 4

ON-GOING SYSTEM MAINTENANCE

A number of system maintenance functions must be performed on a periodic schedule. Backups of system files on a regular basis is necessary to prevent data destruction or loss in the event of a computer failure. Many of the files require periodic reorganization for efficient operation and to prevent system failure. Several directory and report files have to be generated periodically to be available to users.

System Backup Procedures

The following table provides a suggested schedule for file backups. All files should be backed up at least yearly and the backup tape physically removed from the computer site.

File	After Major Update	Every 2 Weeks	Every Month	Every 3 Months	Every 6 Months	Annual Off-Site
HIS.OBJECT		x				x
HIS.LOADTST		x				x
HIS.SUBRTN4		x				x
HIS.REL4PTØ		x				x
HIS.TABLES		x				x
HIS.ACCIDENT	x		x			x
HIS.ACCVEH	x		x			x
HIS.BRIDGE	x				x	x
HIS.RAILROAD	x				x	x
HIS.ROADLOG	x			x		x
HIS.SKID	x			x		x
HIS.SUFFICY	x					x
HIS.TRAFFIC	x			x		x
HIS.TRUMILE	x			x		x
HIS.USNØ52	x		x			x
HIS.GRIDTBL	x					x
HIS.DEFENSE	x					x
HIS.SIGNEDT	x					x
HIS.CTYXØ52	x					x
HIS.MAINTDIV	x					x
HIS.MAINTSEC	x					x

The System Libraries - The system libraries should be backed up at least once every two weeks. The IBM utility IEHMOVE can be used for backing up the libraries and for restoring them from a backup when necessary. The following job setup can be used to copy the libraries to a backup tape:

```
// JOB
// EXEC PGM=IEHMOVE
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DISP=OLD,UNIT=SYSDA,VOL=SER=SPOOL1
//DD1    DD DISP=OLD,UNIT=SYSDA,VOL=SER=WORKØ1
//TAPE   DD DISP=(NEW,PASS),UNIT=TAPE,VOL=SER=serial,DSNAME=name,
//          DCB=(BLKSIZE=8ØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
  COPY  PDS=HIS.OBJECT,TO=TAPE=(serial,1),TODD=TAPE
  COPY  PDS=HIS.SUBRTN4,TO=TAPE=(serial,2),TODD=TAPE
  COPY  PDS=HIS.LOADTST,TO=TAPE=(serial,3),TODD=TAPE
  COPY  PDS=HIS.REL4PTØ,TO=TAPE=(serial,4),TODD=TAPE
  COPY  PDS=HIS.TABLES,TO=TAPE=(serial,5),TODD=TAPE
/*
```

The system's source modules are stored in the PANVALET system of the Department of Highways. The PANVALET system is periodically copied to tape by the Department of Highways.

The Accident Detail and Vehicle Files - These files have a high update rate, since records are added each time a traffic accident occurs. They should be copied to tape and reorganized once a month. The backup should be timed to take place after a major update of the files. To back up the files and to reorganize the files, submit the following job setup:

```
// JOB
// EXEC HISACC,DISP=OLD
//SAVEACC DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name
//SYSIN DD *
:REORGANIZE,FILE=ACCIDENT,PASSWORD=password
/*
```

It is not necessary to specify DCB information (record length or block size) in the SAVEACC DD statement. To back up the files without reorganiz ing,replace the name REORGANIZE with the name COPY in the above job setup.

If it is necessary to restore the accident files from a backup file, submit the following job setup:

```
// JOB
// EXEC HISACC,DISP=OLD
//SAVEACC DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=ACCIDENT,PASSWORD=password
/*
```

The Bridge File - The bridge file has a low update rate. The file should be backed up about once every 6 months, but should be backed up after any major updates. Use the following job setup to back up the bridge file:

```
// JOB
// EXEC HISBRID,DISP=OLD
//SAVEBDG DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=268ØØ,LRECL=268,RECFM=FB)
//SYSIN DD *
:COPY,FILE=BRIDGE,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. If it is necessary to restore the file from a backup file, submit the following job setup:

```
// JOB
// EXEC HISBRID,DISP=OLD
//SAVEBDG DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=BRIDGE,PASSWORD=password
/*
```

The Railroad File - The railroad file has a low update rate. The file should be backed up about once every 6 months, but should be backed up after any major updates. Use the following job setup to back up the railroad file:

```
// JOB
// EXEC HISRRX,DISP=OLD
//SAVERRX DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=185ØØ,LRECL=185,RECFM=FB)
//SYSIN DD *
:COPY,FILE=RAILROAD,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. If it is necessary to restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISRRX,DISP=OLD
//SAVERRX DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:COPY,FILE=RAILROAD,PASSWORD=password
/*
```

The Roadlog File - The roadlog file has a moderate update rate. It should be backed up about once every three months. It should also be backed up immediately following any major update. It may be desirable to combine a reorganization with the backup. The PRINT-DSCB program can be run to help determine whether reorganization is necessary. To backup without reorganizing, submit the following run:

```
// JOB
// EXEC HISRLG,DISP=OLD
//SAVERLG DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=16ØØØ,LRECL=16Ø,RECFM=FB)
//SYSIN DD *
:COPY,FILE=ROADLOG,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISRLG,DISP=OLD
//SAVERLG DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=ROADLOG,PASSWORD=password
/*
```

The traffic and true mileage files should be backed up whenever the roadlog file is backed up. It is essential to system operation that these files be in agreement with each other at all times.

The Skid File - The update rate of the skid file varies during the year. During the warmer months the file may have a higher update rate than during the colder months. In general, the file should be backed up about once every three months as well as after any major update. The file will probably need to be reorganized during at least every other backup. The PRINT-DSCB program can aid in determining when reorganization is needed. To back up the file without reorganizing, submit the following job setup:

```
// JOB
// EXEC HISSKID,DISP=OLD
//SAVESKD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=172ØØ,LRECL=86,RECFM=FB)
//SYSIN DD *
:REORGANIZE,FILE=SKID,TYPE-RUN=BACKUP,PASSWORD=password
/*
```

To reorganize during the backup, change TYPE-RUN=BACKUP to TYPE-RUN=REORGANIZE. To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISSKID,DISP=OLD
//SAVESKD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:REORGANIZE,FILE=SKID,TYPE-RUN=LOAD,PASSWORD=password
/*
```

The Sufficiency File - The sufficiency file is generally updated once a year to produce the annual sufficiency report. Because of this, an annual backup is adequate for this file. To backup the file, submit the following run:

```
// JOB
// EXEC HISSUFF,DISP=OLD
//SAVESUF DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=68ØØ,LRECL=68,RECFM=FB)
//SYSIN DD *
:COPY,FILE=SUFFICIENCY,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISSUFF,DISP=OLD
//SAVESUF DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=SUFFICIENCY,PASSWORD=password
/*
```

The Traffic File - At the first of each year, each record in the traffic file is updated. The UPDATE-BY-YEAR program shifts the data in each record to make room for another year of data. Then the rewrite function of UPDATE is used to enter the new traffic counts. An annual backup should be performed after these operations have taken place. During the year, a number of insertions, deletions, and new-key operations will be submitted to keep the file in agreement with the roadlog and trumile files. A backup of the traffic file should be performed whenever these files are backed up. To backup the file, submit the following run:

```
// JOB
// EXEC HISTRAF,DISP=OLD
//SAVETRF DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=8ØØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
:COPY,FILE=TRAFFIC,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISTRAF,DISP=OLD
//SAVETRF DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=TRAFFIC,PASSWORD=password
/*
```

The True Mileage File - The true mileage file should be backed up whenever the roadlog and traffic files are backed up. Care must be taken so that these three files remain in agreement with each other. To copy the true mileage file, submit the following job setup:

```
// JOB
// EXEC HISTRAF,DISP=OLD
//SAVETRM DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=156ØØ,LRECL=26,RECFM=FB)
//SYSIN DD *
:COPY,FILE=TRUMILE,PASSWORD=password
/*
```

If the file needs reorganization, replace the name COPY with the name REORGANIZE. To restore the file from a backup copy, submit the following run:

```
// JOB
// EXEC HISTRAF,DISP=OLD
//SAVETRM DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE,FILE=TRUMILE,PASSWORD=password
/*
```

The Urban Sign Files - Each city using the urban sign subsystem has a separate file allocated for it. The update rate can be expected to vary from city to city, but as a rule plan to backup each urban sign file once a month. These files will require frequent reorganization. Currently, only Great Falls is utilizing the urban sign software. To backup the Great Falls file, submit the following job setup:

```
// JOB
// EXEC HISUSN,DISP=OLD
//ddname  DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=184ØØ,LRECL=92,RECFM=FB)
//SYSIN DD *
:REORGANIZE,CITY=GREAT-FALLS,TYPE-RUN=BACKUP,DDNAME=ddname,
:  PASSWORD=password
/*
```

If the file needs reorganization, replace TYPE-RUN=BACKUP with TYPE-RUN=REORGANIZE. To restore the file from a backup copy, submit the following run:

```
// JOB
// EXEC HISUSN,DISP=OLD
//ddname  DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:REORGANIZE,CITY=GREAT-FALLS,TYPE-RUN=LOAD,DDNAME=ddname,
:  PASSWORD=password
/*
```

To backup and restore files for other cities, change CITY=GREAT-FALLS to CITY=city, where "city" is the city name. A list of city names is provided in table 2-1 of the user's manual.

The Grid Table - The grid table has a low update rate. An annual backup will normally be sufficient. A backup should be performed after any major update. To back up the file, submit the following run:

```
// JOB
// EXEC HIS
//INDD  DD DISP=SHR,DSNAME=HIS.GRIDTBL
//OUTDD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//         DISP=(NEW,KEEP),DSNAME=name,
//         DCB=(BLKSIZE=8ØØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

To restore the file from a backup copy, submit the following run:

```
// JOB
// EXEC HIS
//INDD  DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//         DISP=OLD,DSNAME=name
//OUTDD DD DISP=OLD,DSNAME=HIS.GRIDTBL
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

The Defense Cross-Reference File - The defense cross-reference file has a low update rate. It should be backed up annually after the annual updates. To back up the file, submit the following run:

```
// JOB
// EXEC HIS
//INDD  DD DISP=SHR,DSNAME=HIS.DEFENSE
//OUTDD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//         DISP=(NEW,KEEP),DSNAME=name,
//         DCB=(BLKSIZE=8ØØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//         DISP=OLD,DSNAME=name
//OUTDD DD DISP=OLD,DSNAME=HIS.DEFENSE
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

The Sign Code Cross-reference File - The sign code cross-reference file has a low update rate. An annual backup will normally be sufficient. To backup the file, submit the following run:

```
// JOB
// EXEC HISUSN,DISP=OLD
//SAVESXR DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=8ØØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
:SIGN-CODE-XREF,TYPE-RUN=STORE,PASSWORD=password
/*
```

If the file needs reorganization, add a second command after the above SIGN-CODE-XREF command that specifies SIGN-CODE-XREF,TYPE-RUN=RESTORE. To restore the file from a backup file, submit the following job setup:

```
// JOB
// EXEC HISUSN,DISP=OLD
//SAVESXR DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:SIGN-CODE-XREF,TYPE-RUN=RESTORE,PASSWORD=password
/*
```

The City Route Cross-Reference Files - Each city that prepares a city route cross-reference file has a separate file. Each file needs to be backed up annually as well as after any major update. To backup a city route cross-reference file, submit the following job setup:

```
// JOB
// EXEC HISUSN,DISP=OLD
//SAVECTY DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//           DISP=(NEW,KEEP),DSNAME=name,
//           DCB=(BLKSIZE=39ØØ,LRECL=39,RECFM=FB)
//SYSIN DD *
:CITY-ROUTE-XREF,CITY=GREAT-FALLS,TYPE-RUN=STORE,PASSWORD=password
/*
```

If the file needs reorganization, follow the above command with a second command that specifies TYPE-RUN=RESTORE. To restore a file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HISUSN,DISP=OLD
//SAVECTY DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//           DISP=OLD,DSNAME=name
//SYSIN DD *
:CITY-ROUTE-XREF,CITY=GREAT-FALLS,TYPE-RUN=RESTORE,PASSWORD=password
/*
```

At the present time, Great Falls is the only city that has a city route cross-reference file. Other cities are expected to add files as time goes by. To backup another city, change CITY=GREAT-FALLS to CITY=city, where "city" is the city name.

The Maintenance Division Cross-Reference File - This file has a low update rate. An annual backup will be satisfactory. To backup the file, submit the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD DISP=SHR,DSNAME=HIS.MAINTDIV
//OUTDD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//         DISP=(NEW,KEEP),DSNAME=name,
//         DCB=(BLKSIZE=86ØØ,LRECL=43,RECFM=FB)
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//         DISP=OLD,DSNAME=name
//OUTDD DD DISP=OLD,DSNAME=HIS.MAINTDIV
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

The Maintenance Section Cross-Reference File - This file has a low update rate. An annual backup should prove sufficient. To backup the file, submit the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD DISP=SHR,DSNAME=HIS.MAINTSEC
//OUTDD DD UNIT=TAPE,VOL=SER=serial,LABEL=(sequence,RETPD=retpd),
//         DISP=(NEW,KEEP),DSNAME=name,
//         DCB=(BLKSIZE=8ØØØ,LRECL=8Ø,RECFM=FB)
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

To restore the file from a backup copy, submit the following job setup:

```
// JOB
// EXEC HIS
//INDD  DD UNIT=TAPE,VOL=SER=serial,LABEL=sequence,
//         DISP=OLD,DSNAME=name
//OUTDD DD DISP=OLD,DSNAME=HIS.MAINTSEC
//SYSIN DD *
:COPY-SEQL-FILE
/*
```

File Reorganization

The following table provides a suggested schedule for file reorganization:

File	After Major Update	Every 2 Weeks	Every Month	Every 3 Months	Every 6 Months	Annual
HIS.OBJECT		x				
HIS.LOADTST		x				
HIS.SUBRTN4			x			
HIS.REL4PTØ			x			
HIS.TABLES			x			
HIS.ACCIDENT			x			
HIS.ACCVEH			x			
HIS.BRIDGE						x
HIS.RAILROAD						x
HIS.ROADLOG	x				x	
HIS.SKID	x				x	
HIS.SUFFICY						x
HIS.TRAFFIC	x				x	
HIS.TRUMILE	x				x	
HIS.USNØ52	x		x			
HIS.SIGNEDT	x					x
HIS.CTYXØ52	x					x

It is largely a matter of personal judgment when to reorganize a file. The PRINT-DSCB program can help determine when an indexed-sequential file needs reorganization. The PRINT-DSCB program is described earlier in this publication in chapter 3.

To reorganize a library, use the COMPRESS procedure provided by the Montana Department of Highways. For example, to compress HIS.TABLES use the job setup:

```
// JOB
// EXEC COMPRESS,LIBRARY='HIS.TABLES'
```

Information on reorganizing the indexed-sequential files is included in the previous section, "System Backup Procedures."

Directory and Report Files

A number of directory and report files are periodically generated from the system files. Some of these (such as the accident directory file) are used a great deal and must be generated frequently. Others are used only for producing annual reports and need to be generated annually. The following table provides a suggested schedule for generating these files:

File	Monthly	Annual
HIS.ACCDIRI	x	
HIS.ACCSECT		x
HIS.BDGREP		x
HIS.RRXREP		x
HIS.TRAFREP		x
HIS.SUFFREP		x

The Accident Directory File - The accident directory file gets quite a lot of use, and should be generated monthly. To generate the file, submit the following run:

```
// JOB
// EXEC HISACD,DISP=OLD
//SYSIN DD *
:CREATE-FA-ACC-DIREC,PASSWORD=password
/*
```

When generating the accident-by-sections file, a slightly different version of the directory file is needed. Accidents that occurred in PTW sections must be shown on the interstate system rather than on the primary system. To generate this version of the file, first check to be sure that the PTW table is up to date. Allocate a file for this version of the directory file so that other users will not be disturbed, then submit the following run:

```
// JOB
// EXEC HISACD
//ACCDIRI DD DISP=OLD,DSNAME=name
//SYSIN DD *
:CREATE-FA-ACC-DIREC,PTW-CONVERT=YES ,PASSWORD=password
/*
```

The following parameters are available for use on CREATE-FA-ACC-DIREC commands. They have little use except as debugging aids.

DATA
START-MILEPOINT
END-MILEPOINT
MAX-#-ENTRIES

The Accident-by-Sections File - The accident-by-sections file is used to print the annual accident-by-sections report. Before generating this file, be sure the following have been done:

1. Generate a separately-allocated version of the accident directory file specifying PTW-CONVERT=YES.
2. Check to be sure that the accident-by-sections table is up-to-date.
3. Check to be sure that the roadlog, traffic, and true mileage files are in agreement.
4. Check to be sure that an up-to-date version of the traffic report file is stored.

The accident-by-sections is then generated by using the following job setup:

```
// JOB
// EXEC HISACCA,DISP=OLD
//ACCDIRI DD DISP=SHR,DSNAME=name
//SYSIN DD *
:CREATE-ACC-BY-SECTN,ACCIDENT-LEVEL=ALL/REPORTABLE/NOT-REPORTABLE
/*
```

If ACCIDENT-LEVEL=ALL is specified, all accidents are included. If ACCIDENT-LEVEL=REPORTABLE is specified, only legally reportable accidents are included. If ACCIDENT-LEVEL=NOT-REPORTABLE is specified, only those accidents that are not legally reportable are included. The generated file will contain data for the interstate and primary systems. DATA, START-MILEPOINT, and END-MILEPOINT may be coded on the command to specify a different data range. The parameter DEBUG=Ø/3/5 may be coded. DEBUG=Ø requests no debug output. DEBUG=3 requests a listing of data used in generating the file. DEBUG=5 requests the listing of data used and also specifies that a user-1ØØ abend is to be issued in the event of an error.

The CREATE-ACC-BY-SECTN operates in two passes. During the first pass (the sections phase), a basic file is constructed that defines the beginning and ending of each section. During the second pass (the record phase), the file is completed by filling in roadlog, traffic, and accident data. The program can be instructed to execute only the first pass by specifying CHECKPOINT=SECTIONS on the command. If the first pass has been successfully executed, the program can be instructed to execute only the second pass by specifying RESTART=RECORD on the command. The parameter LIST=ERROR/FINAL/ALL can be coded on the command. LIST=ERROR requests a listing of the file only if an error occurs during execution. LIST=FINAL requests a listing of the file during the second pass. LIST=ALL requests a listing of the file during both passes.

The Bridge Report File - The bridge report file is needed annually for producing the bridge defense listing and the pre-attack bridge tape. Before attempting to generate the bridge report file, be sure that the defense cross-reference file is up-to-date. The file is generated with the following job setup:

```
// JOB
// EXEC HISBRID,DISP=OLD
//SYSIN DD *
:CREATE-BDGREP,PASSWORD=password
/*
```

The program's execution consists of a step that generates the file followed by a step that sorts the file into the proper order. If the sort fails, the program can be restarted at the sort step by specifying RESTART=SORT on the command. The MAX-#-ENTRIES parameter can be coded on the command to limit the number of bridges processed (this parameter is useful only during program debugging).

The Railroad Report File - The railroad report file is needed annually for producing the railroad report. The file is generated with the following job setup:

```
// JOB
// EXEC HISRRX,DISP=OLD
//SYSIN DD *
:CREATE-RRXREP,PASSWORD=password
/*
```

The Traffic Report File - This file is generally needed annually. It is used for producing the annual traffic-by-sections report. It also has a number of other uses throughout HIS, such as in generating the accident-by-sections file and in generating the sufficiency report file. For this reason, it may need to be generated on an as-needed basis throughout the year. To create the file, first be sure that the roadlog, traffic, and true mileage files are in agreement. Then submit the following job setup:

```
// JOB
// EXEC HISTRAF,DISP=OLD
//SYSIN DD *
:CREATE-TRAFREP ,PASSWORD=password
/*
```

The following parameters can be coded on CREATE-TRAFREP commands:

DATA

START-MILEPOINT

END-MILEPOINT

The Sufficiency Report File - This file is needed for the annual sufficiency report. Before generating this file, check the following:

1. The roadlog, traffic, true mileage, and sufficiency files must all be in agreement.
2. An up-to-date version of the traffic report file must be stored.
3. An up-to-date version of the accident directory file must be stored.

The sufficiency report file is generated with the following job setup:

```
// JOB
// EXEC HISSUFF,DISP=OLD
//SYSIN DD *
:CREATE-SUFFREP,PASSWORD=password
/*
```

The following parameters can be coded on CREATE-SUFFREP commands:

DATA,START-MILEPOINT,END-MILEPOINT (default - DATA=PRIMARY)

LIST=ERROR/FINAL/ALL

DEBUG=Ø/5

CHECKPOINT=SUFFICIENCY/ROADLOG/TRAFFIC/ACCIDENT

RESTART=ROADLOG/TRAFFIC/ACCIDENT/CALCULATIONS

DEBUG=5 specifies that a user-1ØØ abend is issued if an error is detected.

The CREATE-SUFFREP program operates in 6 passes:

1. Sufficiency phase - Builds basic file with sufficiency data.
2. Roadlog phase - Adds roadlog data to file.
3. Traffic phase - Adds traffic data to file.
4. Accident phase - Adds accident data to file.
5. Calculations phase - Calculates sufficiency ratings based on information stored by passes 1-4.
6. Load phase - Stores the final indexed-sequential file.

LIST=ERROR requests a file listing only if an error occurs. LIST=FINAL requests a file listing during the load phase. LIST=ALL requests a file listing during each phase. The CHECKPOINT parameter can be used to terminate the program after a specified pass. The RESTART parameter can be used to restart the program at a specified pass (all previous passes must have been successfully completed).

CHAPTER 5

FILE ALLOCATIONS

This chapter provides the information needed to allocate the various Highway Information System files.

The Accident Detail File

Organization: Indexed-sequential.

Item	Value
DDNAME	ACIDENT
DSNAME	HIS.ACCIDENT
RECFM	FB
LRECL	96
KEYLEN	12
RKP	1
OPTCD	LIYR
CYLOFL	2

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,96Ø	135	135	216Ø
333Ø	2	6,336	66	132	2112
333Ø	3	4,128	43	129	2Ø64
* 333Ø	4	3,Ø72	32	128	2Ø48
333Ø	5	2,4ØØ	25	125	2ØØØ
333Ø	6	1,92Ø	2Ø	12Ø	192Ø
333Ø	7	1,632	17	119	19Ø4
333Ø	8	1,44Ø	15	12Ø	192Ø
333Ø	9	1,248	13	117	1872
333Ø	1Ø	1,Ø56	11	11Ø	176Ø
2314	1	7,2ØØ	75	75	1275
2314	2	3,456	36	72	1224
2314	3	2,2Ø8	23	69	1173
2314	4	1,632	17	68	1156
2314	5	1,248	13	65	11Ø5
2314	6	96Ø	1Ø	6Ø	1Ø2Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=2.

Current Allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	5Ø	333Ø dual density	333ØØ2
Overflow	9	333Ø dual density	333ØØ2

Current file capacity: 1Ø2,4ØØ records.

Approximate number of records as of February 1976: 75,ØØØ.

Approximate number of records added each year: 2Ø,ØØØ.

The Accident Vehicle File

Organization: Indexed-sequential.

Item	Value
DDNAME	ACCVEH
DSNAME	HIS.ACCVEH
RECFM	FB
LRECL	136
KEYLEN	15
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,92Ø	95	95	1,425
333Ø	2	6,256	46	92	1,38Ø
* 333Ø	3	4,Ø8Ø	3Ø	9Ø	1,35Ø
333Ø	4	2,992	22	88	1,32Ø
333Ø	5	2,312	17	85	1,275
333Ø	6	1,9Ø4	14	84	1,26Ø
333Ø	7	1,632	12	84	1,26Ø
333Ø	8	1,36Ø	1Ø	8Ø	1,2ØØ
333Ø	9	1,224	9	81	1,215
333Ø	1Ø	1,Ø88	8	8Ø	1,2ØØ
2314	1	7,2Ø8	53	53	848
2314	2	3,4ØØ	25	5Ø	8ØØ
2314	3	2,176	16	48	768
2314	4	1,632	12	48	768
2314	5	1,224	9	45	72Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	113	333Ø dual density	333ØØ2
Overflow	16	333Ø dual density	333ØØ2
	13Ø		

Current file capacity: 152,55Ø records.

Approximate number of records as of February 1976: 125,ØØØ.

Approximate number of records added each year: 35,ØØØ.

The Accident Directory File

Organization: Indexed-sequential.

Item	Value
DDNAME	ACCDIRI
DSNAME	HIS.ACCDIRI
RECFM	FB
LRECL	44
KEYLEN	25
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
* 333Ø	1	12,936	294	294	4,41Ø
333Ø	2	6,336	144	288	4,32Ø
333Ø	3	4,136	94	282	4,23Ø
333Ø	4	3,Ø36	69	276	4,14Ø
333Ø	5	2,376	54	27Ø	4,Ø5Ø
333Ø	6	1,936	44	264	3,96Ø
333Ø	7	1,628	37	259	3,885
333Ø	8	1,4Ø8	32	256	3,84Ø
333Ø	9	1,232	28	252	3,78Ø
333Ø	1Ø	1,1ØØ	25	25Ø	3,75Ø
2314	1	7,216	164	164	2,624
2314	2	3,432	78	156	2,496
2314	3	2,2ØØ	5Ø	15Ø	2,4ØØ
2314	4	1,584	36	144	2,3Ø4
2314	5	1,232	28	14Ø	2,24Ø

* Current allocation

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	1Ø	333Ø dual density	333ØØ2
Overflow	4	333Ø dual density	333ØØ2
	15		

Current file capacity: 44,1ØØ records.

Approximate number of records as of February 1976: 3Ø,ØØØ.

Approximate number of records added each year: 1Ø,ØØØ.

The Accident-by-Sections File

Organization: Sequential.

Item	Value
DDNAME	ACCSECT
DSNAME	HIS.ACCSECT
RECFM	FB
LRECL	146

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,994	89	89	1,691
333Ø	2	6,424	44	88	1,672
333Ø	3	4,234	29	87	1,653
333Ø	4	3,Ø66	21	84	1,596
* 333Ø	5	2,482	17	85	1,615
333Ø	6	2,Ø44	14	84	1,596
333Ø	7	1,6Ø6	11	77	1,463
333Ø	8	1,46Ø	1Ø	8Ø	1,52Ø
333Ø	9	1,314	9	81	1,539
333Ø	1Ø	1,168	8	8Ø	1,52Ø
2314	1	7,154	49	49	98Ø
2314	2	3,5Ø4	24	48	96Ø
2314	3	2,19Ø	15	45	9ØØ
2314	4	1,6Ø6	11	44	88Ø
2314	5	1,314	9	45	9ØØ
2314	6	1,Ø22	7	42	84Ø
2314	7	876	6	42	84Ø
2314	8	73Ø	5	4Ø	8ØØ

* Current allocation.

Current allocation:

Cylinders	Unit Type	Volume
6	333Ø dual density	333ØØ2

Current file capacity: 9,69Ø records.

Approximate number of records as of February 1976: 6,ØØØ.

Note: This file will normally contain about the same number of records as the traffic report file.

The Accident Memos File

Organization: Sequential.

Item	Value
DDNAME	MEMOIN,MEMOOUT
DSNAME	HIS.MEMOS
RECFM	FB
LRECL	9Ø
BLKSIZE	117Ø

Current allocation:

Cylinders	Unit Type	Volume
6	333Ø dual density	333ØØ2

Current file capacity: 14,72Ø.

Number of records in file: Varies.

The Bridge File

Organization: Indexed-sequential.

Item	Value
DDNAME	BRIDGE
DSNAME	HIS.BRIDGE
RECFM	FB
LRECL	268
KEYLEN	16
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
* 333Ø	1	12,864	48	48	72Ø
333Ø	2	6,164	23	46	69Ø
333Ø	3	4,Ø2Ø	15	45	675
333Ø	4	2,948	11	44	66Ø
333Ø	5	2,412	9	45	675
333Ø	6	1,876	7	42	63Ø
333Ø	7	1,6Ø8	6	42	63Ø
2314	1	6,968	26	26	416
2314	2	3,216	12	24	384
2314	3	2,144	8	24	384
2314	4	1,6Ø8	6	24	384
2314	5	1,Ø72	4	2Ø	32Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=3

Current Allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	8	333Ø dual density	333ØØ2
Overflow	2	333Ø dual density	333ØØ2
	11		

Current file capacity: 5,768 records.

Approximate number of records as of February 1976: 3,5ØØ.

The Bridge Report File

Organization: Sequential.

Item	Value
DDNAME	BDGREP
DSNAME	HIS.BDGREP
RECFM	FB
LRECL	1Ø4
BLKSIZE	6344

Current allocation:

Cylinders	Unit Type	Volume
3	333Ø dual density	333ØØ2

Current file capacity: 6,954 records.

The Railroad File

Organization: Indexed-sequential.

Item	Value
DDNAME	RAILROAD
DSNAME	HIS.RAILROAD
RECFM	FB
LRECL	185
KEYLEN	15
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,95Ø	7Ø	7Ø	1,19Ø
333Ø	2	6,29Ø	34	68	1,156
333Ø	3	4,Ø7Ø	22	66	1,122
333Ø	4	2,96Ø	16	64	1,Ø88
333Ø	5	2,4Ø5	13	65	1,1Ø5
333Ø	6	1,85Ø	1Ø	6Ø	1,Ø2Ø
333Ø	7	1,665	9	63	1,Ø71
2314	1	7,215	39	39	7Ø2
2314	2	3,33Ø	18	36	648
* 2314	3	2,22Ø	12	36	648
2314	4	1,48Ø	8	32	576
2314	5	1,11Ø	6	3Ø	54Ø
2314	6	925	5	3Ø	54Ø

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	2314	23141Ø
Prime	4	2314	23141Ø
Overflow	1	2314	23141Ø
	6		

Current file capacity: 2,592 records.

Approximate number of records as of February 1976: 1ØØØ.

The Railroad Report File

Organization: Indexed-sequential.

Item	Value
DDNAME	RRREP
DSNAME	HIS.RRREP
RECFM	FB
LRECL	95
KEYLEN	15
RKP	1
OPTCD	LYR
CYLOFL	1

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,92Ø	136	136	2,312
333Ø	2	6,365	67	134	2,278
333Ø	3	4,18Ø	44	132	2,244
333Ø	4	3,Ø4Ø	32	128	2,176
333Ø	5	2,375	25	125	2,125
2314	1	7,22Ø	76	76	1,368
2314	2	3,42Ø	36	72	1,296
2314	3	2,185	23	69	1,242
2314	4	1,615	17	68	1,224
2314	5	1,235	13	65	1,17Ø
* 2314	6	95Ø	1Ø	6Ø	1,Ø8Ø

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=1.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	2314	23141Ø
Prime	3	2314	23141Ø

Current file capacity: 2,88Ø records.

Approximate number of records as of February 1976: 1,ØØØ.

The Roadlog File

Organization: Indexed-sequential.

Item	Value
DDNAME	ROADLOG
DSNAME	HIS.ROADLOG
RECFM	FB
LRECL	16Ø
KEYLEN	15
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,8ØØ	8Ø	8Ø	1,2ØØ
333Ø	2	6,24Ø	39	78	1,17Ø
333Ø	3	4,16Ø	26	78	1,17Ø
333Ø	4	3,Ø4Ø	19	76	1,14Ø
333Ø	5	2,4ØØ	15	75	1,125
333Ø	6	1,92Ø	12	72	1,Ø8Ø
333Ø	7	1,6ØØ	1Ø	7Ø	1,Ø5Ø
2314	1	7,2ØØ	45	45	72Ø
* 2314	2	3,36Ø	21	42	672
2314	3	2,Ø8Ø	13	39	624
2314	4	1,6ØØ	1Ø	4Ø	64Ø
2314	5	1,12Ø	7	35	56Ø
2314	6	96Ø	6	36	576

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	2314	23141Ø
Prime	15Ø	2314	23141Ø
Overflow	9	2314	23141Ø
	16Ø		

Current file capacity: 1ØØ,8ØØ records.

Approximate number of records as of February, 1976: 8Ø,ØØØ.

The Skid File

Organization: Indexed-sequential.

Item	Value
DDNAME	SKID
DSNAME	HIS.SKID
RECFM	FB
LRECL	86
KEYLEN	28
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

	Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
	333Ø	1	12,9ØØ	15Ø	15Ø	2,25Ø
	333Ø	2	6,278	73	146	2,19Ø
	333Ø	3	4,128	48	144	2,16Ø
	333Ø	4	3,Ø1Ø	35	14Ø	2,1ØØ
	333Ø	5	2,4Ø8	28	14Ø	2,1ØØ
	333Ø	6	1,892	22	132	1,98Ø
	333Ø	7	1,634	19	133	1,995
	333Ø	8	1,376	16	128	1,92Ø
*	2314	1	7,138	83	83	1,328
	2314	2	3,44Ø	4Ø	8Ø	1,28Ø
	2314	3	2,15Ø	25	75	1,2ØØ
	2314	4	1,548	18	72	1,152
	2314	5	1,2Ø4	14	7Ø	1,12Ø

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	2314	231432
Prime	3Ø	2314	231432
Overflow	4	2314	231432

Current file capacity: 39,84Ø records.

Approximate number of records as of February 1976: 22,ØØØ.

The Sufficiency File

Organization: Indexed-sequential.

Item	Value
DDNAME	SUFFICY
DSNAME	HIS.SUFFICY
RECFM	FB
LRECL	68
KEYLEN	15
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,92Ø	19Ø	19Ø	2,85Ø
* 333Ø	2	6,324	93	186	2,79Ø
333Ø	3	4,148	61	183	2,745
333Ø	4	3,Ø6Ø	45	18Ø	2,7ØØ
333Ø	5	2,38Ø	35	175	2,625
2314	1	7,2Ø8	1Ø6	1Ø6	1,696
2314	2	3,4ØØ	5Ø	1ØØ	1,6ØØ
2314	3	2,176	32	96	1,536
2314	4	1,632	24	96	1,536
2314	5	1,224	18	9Ø	1,44Ø
2314	6	1,Ø2Ø	15	9Ø	1,44Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	2	333Ø dual density	333ØØ2
Overflow	1	333Ø dual density	333ØØ2

Current file capacity: 5,58Ø records.

Approximate number of records as of February 1976: 2,ØØØ.

The Sufficiency Report File

Organization: Indexed-sequential.

Item	Value
DDNAME	SUFFREP
DSNAME	HIS.SUFFREP
RECFM	FB
LRECL	1Ø9
KEYLEN	15
RKP	1
OPTCD	LYR
CYLOFL	1

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,862	118	118	2,ØØ6
333Ø	2	6,322	58	116	1,972
333Ø	3	4,142	38	114	1,938
333Ø	4	3,Ø52	28	112	1,9Ø4
333Ø	5	2,398	22	11Ø	1,87Ø
* 333Ø	1Ø	1,Ø9Ø	1Ø	1ØØ	1,7ØØ
2314	1	7,194	66	66	1,188
2314	2	3,379	31	62	1,116
2314	3	2,18Ø	2Ø	6Ø	1,Ø8Ø
2314	4	1,526	14	56	1,ØØ8
2314	5	1,199	11	55	99Ø

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=1.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	2	333Ø dual density	333ØØ2

Current file capacity: 3,4ØØ records.

Approximate number of records as of February 1976: 2,ØØØ.

The Traffic File

Organization: Indexed-sequential.

Item	Value
DDNAME	TRAFFIC
DSNAME	HIS.TRAFFIC
RECFM	FB
LRECL	8Ø
KEYLEN	15
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
* 333Ø	1	12,88Ø	161	161	2,415
333Ø	2	6,32Ø	79	158	2,37Ø
333Ø	3	4,16Ø	52	156	2,34Ø
333Ø	4	3,Ø4Ø	38	152	2,28Ø
333Ø	5	2,4ØØ	3Ø	15Ø	2,25Ø
333Ø	6	1,92Ø	24	144	2,16Ø
333Ø	7	1,6ØØ	2Ø	14Ø	2,1ØØ
333Ø	8	1,36Ø	17	136	2,Ø4Ø
2314	1	7,2ØØ	9Ø	9Ø	1,44Ø
2314	2	3,44Ø	43	86	1,376
2314	3	2,16Ø	27	81	1,296
2314	4	1,6ØØ	2Ø	8Ø	1,28Ø
2314	5	1,2ØØ	15	75	1,2ØØ

* Current allocation.

Note: Records/cylinder calculated using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	5	333Ø dual density	333ØØ2
Overflow	2	333Ø dual density	333ØØ2
	8		

Current file capacity: 12,Ø75 records.

Approximate number of records as of February 1976: 6,5ØØ.

The Traffic Report File

Organization: Indexed-sequential.

Item	Value
DDNAME	TRAFREP
DSNAME	HIS.TRAFREP
RECFM	FB
LRECL	96
KEYLEN	15
RKP	1
OPTCD	LYR
CYLOFL	1

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
* 333Ø	1	12,864	134	134	2,278
333Ø	2	6,336	66	132	2,244
333Ø	3	4,128	43	129	2,193
333Ø	4	3,Ø72	32	128	2,176
333Ø	5	2,4ØØ	25	125	2,125
333Ø	6	1,92Ø	2Ø	12Ø	2,Ø4Ø
333Ø	7	1,632	17	119	2,Ø23
333Ø	8	1,344	14	112	1,9Ø4
2314	1	7,2ØØ	75	75	1,35Ø
2314	2	3,456	36	72	1,296
2314	3	2,2Ø8	23	69	1,242
2314	4	1,632	17	68	1,224
2314	5	1,248	13	65	1,17Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=1.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	4	333Ø dual density	333ØØ2
	5		

Current file capacity: 9,112 records.

Approximate number of records as of February 1976: 4,ØØØ.

The True Mileage File

Organization: Indexed-sequential.

Item	Value
DDNAME	TRUMILE
DSNAME	HIS.TRUMILE
RECFM	FB
LRECL	26
KEYLEN	9
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,948	498	498	7,47Ø
333Ø	2	6,37Ø	245	49Ø	7,35Ø
333Ø	3	4,186	161	483	7,245
* 333Ø	4	3,Ø68	118	472	7,Ø8Ø
333Ø	5	2,418	93	465	6,975
333Ø	6	1,976	76	456	6,84Ø
333Ø	7	1,664	64	448	6,72Ø
333Ø	8	1,43Ø	55	44Ø	6,6ØØ
333Ø	9	1,248	48	432	6,48Ø
333Ø	1Ø	1,Ø92	42	42Ø	6,3ØØ
2314	1	7,228	278	278	4,448
2314	2	3,458	133	266	4,256
2314	3	2,236	86	258	4,128
2314	4	1,638	63	252	4,Ø32
2314	5	1,274	49	245	3,92Ø

* Current allocation.

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	4	333Ø dual density	333ØØ2
Overflow	1	333Ø dual density	333ØØ2
	6		

Current file capacity: 28,32Ø records.

Approximate number of records as of February 1976: 14,ØØØ.

Urban Sign Inventory File - Great Falls

Organization: Indexed-sequential.

Item	Value
DDNAME	USNØ52
DSNAME	HIS.USNØ52
RECFM	FB
LRECL	92
KEYLEN	1Ø
RKP	1
OPTCD	LIYR
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,88Ø	14Ø	14Ø	2,1ØØ
* 333Ø	2	6,348	69	138	2,Ø7Ø
333Ø	3	4,14Ø	45	135	2,Ø25
333Ø	4	3,Ø36	33	132	1,98Ø
333Ø	5	2,392	26	13Ø	1,95Ø
333Ø	6	1,932	21	126	1,89Ø
333Ø	7	1,656	18	126	1,89Ø
333Ø	8	1,38Ø	15	12Ø	1,8ØØ
333Ø	9	1,196	13	117	1,755
333Ø	1Ø	1,1Ø4	12	12Ø	1,8ØØ
2314	1	7,176	78	78	1,248
2314	2	3,4Ø4	37	74	1,184
2314	3	2,2Ø8	24	72	1,152
2314	4	1,564	17	68	1,Ø88
2314	5	1,196	13	65	1,Ø4Ø
2314	6	1,Ø12	11	66	1,Ø56

* Current allocation.

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	15	333Ø dual density	333ØØ2
Overflow	4	333Ø dual density	333ØØ2
	2Ø		

Current file capacity: 31,Ø5Ø.

Anticipated number of records by December 1976: 25,ØØØ.

The Grid Table

Organization: Sequential.

Item	Value
DDNAME	GRIDTBL
DSNAME	HIS.GRIDTBL
RECFM	FB
LRECL	8Ø
BLKSIZE	64ØØ

Current allocation:

Cylinders	Unit Type	Volume
2	333Ø dual density	333ØØ2

Current file capacity: 5,Ø4Ø records.

Approximate number of records as of February 1976: 3,ØØØ.

The Defense Cross-Reference File

Organization: Sequential.

Item	Value
DDNAME	DEFENSE,DEFENSEI
DSNAME	HIS.DEFENSE
RECFM	FB
LRECL	8Ø
BLKSIZE	4ØØ

Current allocation:

Cylinders	Unit Type	Volume
1	333Ø dual density	333ØØ2

The Maintenance Division Cross-Reference File

Organization: Sequential.

Item	Value
DDNAME	MAINTDIV
DSNAME	HIS.MAINTDIV
RECFM	FB
LRECL	43
BLKSIZE	6,4Ø7

Current allocation:

Cylinders	Unit Type	Volume
1	333Ø dual density	333ØØ2

The Maintenance Section Cross-Reference File

Organization: Sequential.

Item	Value
DDNAME	MAINTSEC
DSNAME	HIS.MAINTSEC
RECFM	FB
LRECL	8Ø
BLKSIZE	2ØØØ

Current allocation:

Cylinders	Unit Type	Volume
1	333Ø dual density	333ØØ2

The Sign Code Cross-Reference Files

Organization: Indexed-sequential.

Item	Signedt	Sgnxref
DDNAME	SIGNEDT	SGNXREF
DSNAME	HIS.SIGNEDT	HIS.SGNXREF
RECFM	FB	FB
LRECL	8Ø	8Ø
KEYLEN	12	9
RKP	2	14
OPTCD	LYR	LYR
CYLOFL	3	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,948	166	166	2,49Ø
333Ø	2	6,318	81	162	2,43Ø
333Ø	3	4,134	53	159	2,385
333Ø	4	3,Ø42	39	156	2,34Ø
333Ø	5	2,418	31	155	2,325
333Ø	6	1,95Ø	25	15Ø	2,25Ø
333Ø	7	1,638	21	147	2,2Ø5
333Ø	8	1,4Ø4	18	144	2,16Ø

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
2314	1	7,176	92	92	1,472
2314	2	3,432	44	88	1,4Ø8
2314	3	2,184	28	84	1,344
2314	4	1,56Ø	2Ø	8Ø	1,28Ø
2314	5	1,248	16	8Ø	1,28Ø

Note: Records/cylinder computed using CYLOFL=3.

Current allocation:

Signedt:

Type	Cylinders	Unit Type	Volume
Index	1	2314	231415
Prime	3	2314	231415

Sgnxref:

Type	Cylinders	Unit Type	Volume
Index	1	2314	231432
Prime	3	2314	231432

The City Route Cross-Reference File - Great Falls

Organization: Indexed-sequential.

Item	Value
DDNAME	CTYXØ52
DSNAME	HIS.CTYXØ52
RECFM	FB
LRECL	39
KEYLEN	7
RKP	1
OPTCD	LRY
CYLOFL	3

Optimum block sizes:

Device	Blk/Trk	Blksize	Rec/Blk	Rec/Trk	Rec/Cyl
333Ø	1	12,948	332	332	4,98Ø
333Ø	2	6,357	163	326	4,89Ø
333Ø	3	4,173	1Ø7	321	4,815
333Ø	4	3,Ø81	79	316	4,74Ø
333Ø	5	2,418	62	31Ø	4,65Ø
333Ø	6	1,989	51	3Ø6	4,59Ø
333Ø	7	1,677	43	3Ø1	4,515
333Ø	8	1,443	37	296	4,44Ø

Current allocation:

Type	Cylinders	Unit Type	Volume
Index	1	333Ø dual density	333ØØ2
Prime	1	333Ø dual density	333ØØ2

Libraries

Organization: Partitioned.

Item	Tables	Object	Loadtst	Rel4ptØ	Subrtn4
DDNAME	TABLES	OBJECT	LOADTST	REL4PTØ	SUBRTN4
DSNAME	HIS.TABLES	HIS.OBJECT	HIS.LOADTST	HIS.REL4PTØ	HIS.SUBRTN4
RECFM	FB	FB	U	U	U
LRECL	8Ø	8Ø	---	---	---
BLKSIZE	4ØØ	4ØØ	---	---	---

Current allocation:

Library	Cylinders	Unit Type	Volume	Directory Blocks
Tables	1Ø	333Ø single density	WORKØ1	1Ø
Object	1Ø	333Ø single density	WORKØ1	15
Loadtst	15	333Ø single density	WORKØ1	2Ø
Rel4ptØ	3Ø	333Ø single density	WORKØ1	45
Subrtn4	5	333Ø single density	WORKØ1	2Ø

Summary of Disk Storage Use

Volume	Unit Type	Cylinders	File Name
23141Ø	2314	16Ø	HIS.ROADLOG
23141Ø	2314	6	HIS.RAILROAD
23141Ø	2314	4	HIS.RRREP
23141Ø - total	----------	17Ø	
231415	2314	4	HIS.SIGNEDT
231432	2314	4	HIS.SGNXREF
231432	2314	35	HIS.SKID
231432 - total	----------	39	
	Total 2314 cylinders ---	17Ø	
		4	
		39	
		213	

Volume	Unit Type	Cylinders	File Name
WORKØ1	333Ø single density	1Ø	HIS.TABLES
WORKØ1	333Ø single density	1Ø	HIS.OBJECT
WORKØ1	333Ø single density	15	HIS.LOADTST
WORKØ1	333Ø single density	3Ø	HIS.REL4PTØ
WORKØ1	333Ø single density	5	HIS.SUBRTN4
WORKØ1 - total	----------	7Ø	
333ØØ2	333Ø dual density	6Ø	HIS.ACCIDENT
333ØØ2	333Ø dual density	13Ø	HIS.ACCVEH
333ØØ2	333Ø dual density	15	HIS.ACCDIRI
333ØØ2	333Ø dual density	6	HIS.ACCSECT
333ØØ2	333Ø dual density	6	HIS.MEMOS
333ØØ2	333Ø dual density	11	HIS.BRIDGE
333ØØ2	333Ø dual density	3	HIS.BDGREP
333ØØ2	333Ø dual density	4	HIS.SUFFICY
333ØØ2	333Ø dual density	3	HIS.SUFFREP
333ØØ2	333Ø dual density	8	HIS.TRAFFIC
333ØØ2	333Ø dual density	5	HIS.TRAFREP
333ØØ2	333Ø dual density	6	HIS.TRUMILE
333ØØ2	333Ø dual density	2Ø	HIS.USNØ52
333ØØ2	333Ø dual density	2	HIS.GRIDTBL
333ØØ2	333Ø dual density	1	HIS.DEFENSE
333ØØ2	333Ø dual density	1	HIS.MAINTDIV
333ØØ2	333Ø dual density	1	HIS.MAINTSEC
333ØØ2	333Ø dual density	2	HIS.CTYXØ52
333ØØ2 - total	----------	284	

Total 2314 cylinders - 213

Total 333Ø cylinders - 354